Fujitsu Ten has recently developed data search software that is targeted at audio title searching and is also suitable for handling any text data in list format.

The software is provided with functions for registering, editing, deleting and searching, and for acquiring the search results. It compiles each of various “groups” of data into a separate index containing the “record” (line) positions of the data items, thus enabling construction of a flexible system permitting search objects and output objects (results) to be selected separately. Further, the index component is loaded into the memory, realizing high-speed and memory-saving searches requiring fewer accesses to the disk.

This paper presents an overview of the newly-developed data search software plus its basic performance.

This software is included in the HDD-AVN launched in November 2002, forming part of its “Music Juke” functions.


\section*{Introduction}

The data search method for HDD is aimed at improving data search speed and efficiency. Aiming at high-speed data search, the development of an HDD search software is an important task. This search software must operate using a small amount of memory and be able to assign multiple pieces of information to a single stored data item. The development of such software is essential for enhancing the performance of HDDs.

\subsection*{2.1 Requirements for the data search software}

\begin{itemize}
  \item Must be high-speed
  \item Must operate using a small amount of memory
  \item Must be able to assign multiple pieces of information to a single stored data item
\end{itemize}

\begin{table}[h]
\centering
\begin{tabular}{|c|c|c|}
\hline
Item & Requirement & Description \\
\hline
\hline
Search Speed & High-speed & Fast data search \hline
Memory Usage & Small amount & Low memory consumption \hline
Information Assignment & Multiple pieces & Assign multiple pieces of information \hline
\end{tabular}
\caption{Requirements for the data search software}
\end{table}
3.1 DBMS (database management system)

1) DBMS (database management system)

- Must be provided with functions for storing and editing information
- Must keep the "output objects" (search results) separate from the search objects

<table>
<thead>
<tr>
<th>Table</th>
<th>Table structure</th>
</tr>
</thead>
<tbody>
<tr>
<td>Name</td>
<td>Description</td>
</tr>
<tr>
<td>Table 1</td>
<td>Description 1</td>
</tr>
<tr>
<td>Table 2</td>
<td>Description 2</td>
</tr>
</tbody>
</table>

3.2 Table structure

- Table data
3.3 Index structure

Index data

Table customization

Index scope

High-speed and memory-saving operation

Index structure
3.4 Reason for index structure selected

The reason for selecting the index structure is to enable efficient searching and retrieval of data. The index structure chosen is designed to optimize performance and reduce search time. The selected structure is as follows:

- **Search method**: The search method used is an efficient algorithm that allows for quick access to the desired data. It involves a series of steps that start from the root node and proceed through the hierarchical structure until the target data is reached.

The chosen index structure is illustrated in the diagram below. The diagram shows the hierarchical arrangement of the index nodes, with each level representing a step in the search process. The structure is designed to minimize the number of nodes visited during a search, thereby improving search speed.

![Diagram of index structure](image)

<table>
<thead>
<tr>
<th>Index Structure Details</th>
<th>Item</th>
<th>Item Status</th>
</tr>
</thead>
<tbody>
<tr>
<td>Index Type</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Search Method</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Data Access Efficiency</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

The table above summarizes the key features of the chosen index structure, highlighting its effectiveness in optimizing search operations. The structure is designed to support a wide range of queries, ensuring that data can be retrieved quickly and efficiently.

---

The figure below illustrates the hierarchical arrangement of the index nodes, with each level representing a step in the search process. The structure is designed to minimize the number of nodes visited during a search, thereby improving search speed. The search method used is an efficient algorithm that allows for quick access to the desired data. The chosen index structure is as follows:

- **Search method**: The search method used is an efficient algorithm that allows for quick access to the desired data. It involves a series of steps that start from the root node and proceed through the hierarchical structure until the target data is reached.

The chosen index structure is illustrated in the diagram below. The diagram shows the hierarchical arrangement of the index nodes, with each level representing a step in the search process. The structure is designed to minimize the number of nodes visited during a search, thereby improving search speed.

![Diagram of index structure](image)

<table>
<thead>
<tr>
<th>Index Structure Details</th>
<th>Item</th>
<th>Item Status</th>
</tr>
</thead>
<tbody>
<tr>
<td>Index Type</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Search Method</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Data Access Efficiency</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

The table above summarizes the key features of the chosen index structure, highlighting its effectiveness in optimizing search operations. The structure is designed to support a wide range of queries, ensuring that data can be retrieved quickly and efficiently.

---

The figure below illustrates the hierarchical arrangement of the index nodes, with each level representing a step in the search process. The structure is designed to minimize the number of nodes visited during a search, thereby improving search speed. The search method used is an efficient algorithm that allows for quick access to the desired data. The chosen index structure is as follows:

- **Search method**: The search method used is an efficient algorithm that allows for quick access to the desired data. It involves a series of steps that start from the root node and proceed through the hierarchical structure until the target data is reached.

The chosen index structure is illustrated in the diagram below. The diagram shows the hierarchical arrangement of the index nodes, with each level representing a step in the search process. The structure is designed to minimize the number of nodes visited during a search, thereby improving search speed. The search method used is an efficient algorithm that allows for quick access to the desired data. The chosen index structure is as follows:

![Diagram of index structure](image)

<table>
<thead>
<tr>
<th>Index Structure Details</th>
<th>Item</th>
<th>Item Status</th>
</tr>
</thead>
<tbody>
<tr>
<td>Index Type</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Search Method</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Data Access Efficiency</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

The table above summarizes the key features of the chosen index structure, highlighting its effectiveness in optimizing search operations. The structure is designed to support a wide range of queries, ensuring that data can be retrieved quickly and efficiently.
4.1 Configuration of measuring system

4.2 Measurement method

1) Type A

2) Type B

3) Type C

Preparation of registered data for use in measurement

Performance measurement
Measurement of performance

The measurement of performance is required for various applications, such as HDD-AVN title data management.

1) Registration

Registration is necessary to store the title data in the HDD-AVN system. The system must be able to accurately store and retrieve the data.

2) Search I

Search I is used to find specific titles within the HDD-AVN system. The system must be able to retrieve the data quickly and accurately.

3) Search II

Search II is used to find titles based on specific criteria, such as genre or artist. The system must be able to retrieve the data quickly and accurately.

4) Search III

Search III is used to find titles based on specific keywords. The system must be able to retrieve the data quickly and accurately.

5) Search IV

Search IV is used to find titles based on specific metadata, such as release date or language. The system must be able to retrieve the data quickly and accurately.

6) Deletion

Deletion is used to remove titles from the HDD-AVN system. The system must be able to remove the data accurately and without error.

4.3 Results of performance measurement

The results of the performance measurement are shown in the table below.

<table>
<thead>
<tr>
<th>Function</th>
<th>HDD-AVN title data management</th>
<th>Application to HDD-AVN title data management</th>
</tr>
</thead>
<tbody>
<tr>
<td>Registration</td>
<td>5FBSDI*</td>
<td>5FBSDI***</td>
</tr>
<tr>
<td>Search I</td>
<td>5FBSDI*</td>
<td>5FBSDI***</td>
</tr>
<tr>
<td>Search II</td>
<td>5FBSDI*</td>
<td>5FBSDI***</td>
</tr>
<tr>
<td>Search III</td>
<td>5FBSDI*</td>
<td>5FBSDI***</td>
</tr>
<tr>
<td>Search IV</td>
<td>5FBSDI*</td>
<td>5FBSDI***</td>
</tr>
<tr>
<td>Deletion</td>
<td>5FBSDI*</td>
<td>5FBSDI***</td>
</tr>
</tbody>
</table>

Application to HDD-AVN title data management

The application of the performance measurement results to HDD-AVN title data management is shown in the table above.

1) Play list creation functions

Play list creation functions are used to create play lists based on user preferences.

2) Play list sequencing functions

Play list sequencing functions are used to sequence the play lists based on specific criteria.

The results of the performance measurement are shown in the table above.
3) Sequencing cancellation functions

4) Playlist creation functions

5) Title erasure functions

6) Track information acquisition functions

7) Random acquisition functions

8) Track information change functions

9) Registered data searching functions

Conclusion

Profiles of Writers

Hideki Kitao
Entered the company in 1992. Since then, has been involved in the software development for digital application devices, such as noise suppression and voice processing. Currently in the Software R&D Department of the System R&D Division, A.V.C. Products Group.

Masahiko Higashiyama
Entered the company in 2001. Since then, has been involved in the development of data search middleware. Currently in the Software R&D Department of the System R&D Division, A.V.C. Products Group.

Shinji Fukuda
Entered the company in 1979. Since then, has been involved in the development of vehicle mounted information devices. Currently the Manager of the Software R&D Department of the System R&D Division, A.V.C. Products Group.